Lab 7

Reg. No: 19BCE1209

Name: Gautam Sanjay Wadhwani

Course: CSE4001 Parallel and Distributed Computing

Q1. Sample program for barrier

Code:

```
#include<stdio.h>
#include<stdlib.h>
#include<omp.h>

int main()
{
int n, m, i;
printf("Enter value of n followed by elements of array a: ");
scanf("%d", &n);
int a[n];
for(int i = 0; i < n; i++)
{
scanf("%d", &a[i]);
}
printf("Enter value of m followed by elements of array z: ");
scanf("%d", &m);
double y[m], b[n];
int z[m];
for(int i = 0; i < m; i++)
{
scanf("%d", &z[i]);
}
#pragma omp parallel
```

```
{
#pragma omp for
for(i = 1; i < n; i++)
{
b[i] = (a[i] + a[i-1]) / 2.0;
printf("b[%d] = %f\n", i, b[i]);
}
#pragma omp barrier
#pragma omp for
for(i = 0; i < m; i++)
{
y[i] = sqrt(z[i]);
printf("y[%d] = %f\n", i, y[i]);
}
}
return 0;
}
```

Output:

```
gautam@ubuntu:~$ gcc lab_7_1.c -fopenmp -lm
gautam@ubuntu:~$ ./a.out
Enter value of n followed by elements of array a: 4
1 2 3 4
Enter value of m followed by elements of array z: 3
64 4 196
b[1] = 1.500000
b[2] = 2.500000
b[3] = 3.500000
y[0] = 8.000000
y[1] = 2.000000
y[2] = 14.000000
gautam@ubuntu:~$
```

Q2. Sample program for no wait

Code:

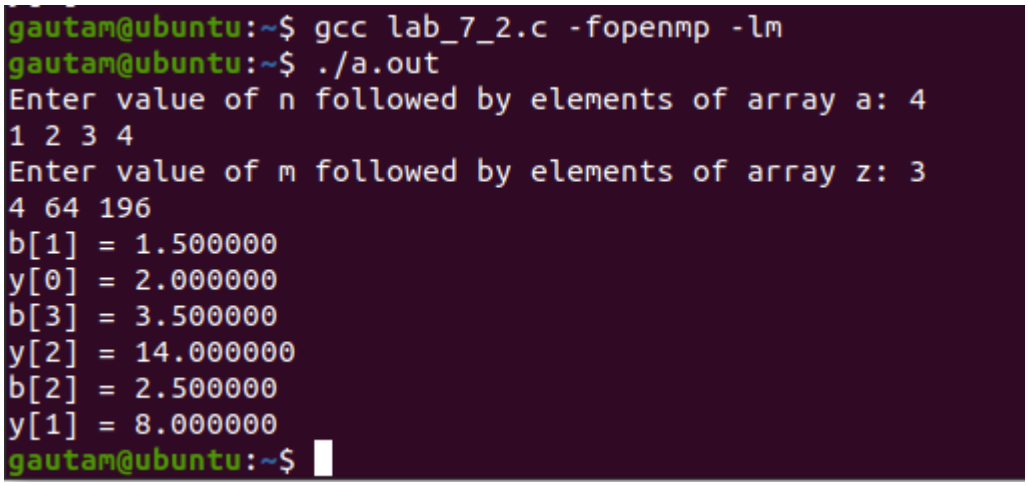
```
#include<stdio.h>
```

```
#include<stdlib.h>
#include<omp.h>

int main()
{
int n, m, i;
printf("Enter value of n followed by elements of array a: ");
scanf("%d", &n);
int a[n];
for(int i = 0; i < n; i++)
{
scanf("%d", &a[i]);
}
printf("Enter value of m followed by elements of array z: ");
scanf("%d", &m);
double y[m], b[n];
int z[m];
for(int i = 0; i < m; i++)
{
scanf("%d", &z[i]);
}
#pragma omp parallel
{
#pragma omp for nowait
for(i = 1; i < n; i++)
{
b[i] = (a[i] + a[i-1]) / 2.0;
printf("b[%d] = %f\n", i, b[i]);
}
#pragma omp for nowait
for(i = 0; i < m; i++)
```

```
{
y[i] = sqrt(z[i]);
printf("y[%d] = %f\n", i, y[i]);
}
}
return 0;
}
```

Output:



Q3. Nested loop parallel for

Code:

```
#include<stdio.h>
#include<stdlib.h>
#include<omp.h>

int main()
{
int n, m;
printf("Enter dimensions of the matrix: ");
scanf("%d %d", &n, &m);
```

```
int a[n][m];
printf("Enter elements of matrix:\n");
for(int i = 0; i < n; i++)
{
for(int j = 0; j < m; j++)
{
scanf("%d", &a[i][j]);
}
}
#pragma omp parallel for
for(int j = 0; j < m; j++)
{
for(int i = 1; i < n; i++)
{
a[i][j] = a[i-1][j] + 2;
printf("a[%d][%d] = %d\n", i, j, a[i][j]);
}
}
for(int i = 0; i < n; i++)
{
for(int j = 0; j < m; j++)
{
printf("%d ", a[i][j]);
}
printf("\n");
}
return 0;
}
```

Output:

```
gautam@ubuntu:~$ gcc lab_7_3.c -fopenmp -lm
gautam@ubuntu:~$ ./a.out
Enter dimensions of the matrix: 4 4
Enter elements of matrix:
0 0 0 0
1 1 1 1
2 2 2 2
3 3 3 3
a[1][0] = 2
a[1][1] = 2
a[2][1] = 4
a[3][1] = 6
a[2][0] = 4
a[1][3] = 2
a[3][0] = 6
a[1][2] = 2
a[2][2] = 4
a[3][2] = 6
a[2][3] = 4
a[3][3] = 6
0 0 0 0
2 2 2 2
4 4 4 4
6 6 6 6
```